Universidad Carlos III de Madrid
e-Archivo

Institutional Repository

This document is published in:

Procedia Computer Science 14 (2012) pp. 254–262

DOI: 10.1016/j.procs.2012.10.029

© 2012. The Authors and Elsevier B.V.

An Approach to User Interface Design of an Accessible User Agent

María González[a,*], Lourdes Moreno[a], Paloma Martínez[a]

[a]*Grupo LaBDA, Departamento de Informática, Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés, Madrid, España*

Abstract: The increase in rich and interactive content on the web has allowed the growth of the number and type of users that access them. This implies that such content should be accessible to anyone, including people with disabilities. Therefore, it should be accompanied with media alternatives (captions, audio description), which should be handled by an accessible user agent that provides support for them. This paper presents: 1) an approach with accessibility requirements following standards to include accessibility in a user agent for delivering accessible multimedia content, and based on it, 2) a conceptualization that provides an abstract user interface model with the ability to be integrated into the development process.

Keywords: Web accessibility; User agent; Media player; Multimedia; Standard; UsiXML.

1. Introduction

Multimedia content, like video, is constantly increasing on the Web. This content has to be accessible through standards like the Web Content Accessibility Guideline (WCAG) [0] of Web Accessibility Initiative (WAI) [0], besides satisfying other accessibility requirements. Additionally, accessible multimedia content on the Web requires that a particular chain of essential, interdependent and accessible components [0] should be considered by user agents. Specifically, media players should enable the delivery of accessible multimedia content in order to obtain a friendly user-video interaction.

[*]Corresponding author. Tel.: +34-916-245-962; fax: +34-916-249-129.
E-mail address: mgonza1@inf.uc3m.es.

Sometimes accessibility barriers inside the user agent that provides video content can cause accessibility problems. For example, this type of user agents or media players could not offer users an intuitive interface to work without a former background or any help; it would make sense to include accessibility requirements if a user who needs to use them cannot access through some assistive technologies or user agents need an specific software but users do not know how to install or use it according to technological diversity of multimedia field.

Taking into account these problems and User Agent Accessibility Guidelines (UAAG) [0] of WAI, our motivation is to design an accessible user agent, which would deliver accessible video content, so as to everybody can access them without any type of barriers.

Regarding what is previously mentioned; we propose a model of accessibility requirements to integrate the design of an accessible user agent in the development process. In order to accomplish that, we use UsiXML (User Interface Extensible Markup Language) [0], a User Interface Description Languages (UIDL) [0], which provides different tools to help the design process of user interfaces through the different abstraction levels.

The remaining of the paper is organized as follows: Section 2 covers related work and background of user agents that provide video content such as standards and regulations. Section 3 introduces a proposal of accessibility requirements for an accessible user agent or media player. Section 4 describes a modeling process with accessibility requirements. And Section 5 gives some conclusions.

2. Background

There are different standards which regulate multimedia content. Among these standards it is found WCAG 2.0 [0], which explains how to make Web content more accessible for people with disabilities. According to its Guideline 1.2, video content shall be accompanied with media alternatives such as captions (or subtitles for hearing handicap people), audio description, sign language, etc. [0]. The WCAG 2.0 guidelines are considered an unofficial standard which is referenced worldwide in most regulations. There are other important initiatives that are totally or partly related to Web accessibility, such as technical standards Section 508, BITV 2, RGAA and AODA. Most of these are less extensive than WCAG but they are very similar. Based on these standards of accessibility content, different checklists have been collected [0-0-0], as well as comparison among them.

As far as multimedia content is concerned, some works can be highlighted [0-0-0] and institutions like [0-0-0-0] that provide good practices guidelines according to accessible multimedia content on the Web.

On the other hand, there are standards which are related to user agent in particular. Among these standards the UAAG of WAI can be outlined. This international standard explains how to make user agents accessible for people with disabilities and how to increase accessibility on Web content. For example, media players shall ensure that all audio and video controls are accessible via keyboard or screen readers. Nowadays, this standard has two versions, UAAG 1.0 (reference version) [0] and UAAG 2.0 (draft version) [0]. This last version gives support to WCAG 2.0. Therefore, according to UAAG 2.0 guidelines, media players should provide support for these media alternatives required by the guideline 1.2 of WCAG 2.0.

Another standard is ISO 9241-171, Ergonomics of human-system interaction [0]. This standard provides guidance on software accessibility. In relation to user agent that provides video content, this international standard includes four guidelines that are necessary in order to consider this type of user agent accessible: 1) user agent shall enable users to stop, start and pause the reproduction; 2) it should enable users to replay, rewind, pause and fast or jump forward the reproduction; 3) it should enable users to select media streams which are presented; 4) it should enable equivalent alternatives to be updated when the content of a media presentation changes. This standard also includes other requirements related to captions, such as, if the contrast between the captions and the background is going to be enough or if the system-wide preferences change during reproduction, the new settings shall be used. Furthermore, the captions position should not interfere the visual content and the captions could be enabled and disabled.

About media player issues, it is necessary to consider the new standard HTML5 (HyperText Markup

Language) [0] which allows users to play videos without installing plug-ins through new labels such as <video> or <audio>. However, HTML5 has some problems caused because it does not support completely the access to the controls provided by Web user agents through keyboard or screen readers. Its current draft version does not give support to include subtitles and audio description, therefore, to solve these problems, it has been included the <track> tag. Nowadays, it does not work properly, but it is still under development up to 2014.

In summary, our goal is to design an accessible user agent that provides video content. Due to this, some related works have been consulted and several media players have been reviewed. Based on an accessibility study of some media players (CCPlayer, BBC iPlayer, Youtube) framed in this research, it was obtained the compliance with some accessibility requirements on the part of the media players. CCPlayer was the most accessible of the three studied media players [0]. Other media players with accessibility features are: JW Player that provides captions and audio description, BSPlayer that provides captions and gives users the possibility of changing the type and the size of the font and VideoLan media player that provides for example keyboard shortcuts and allows users to change size, font or color of the captions among others.

3. Accessibility requirements for a user agent that provides video content

In order to obtain a subset of essential accessibility requirements, it is necessary to make a revision of the standards indicated in Section 2 that covers a wide range of recommendations for providing Web multimedia content in the accessible way such as UAAG 2.0 and ISO 9241-171. This revision has previously been carried out in works like [0-0].

Table 1. User agent accessibility requirements

Code	Name	Description	Group	Subgroup	Source	
					ISO	UAAG 2.0
NR01	Play	Play the video content	Native	Reproduction	10.8.2	2.11.6
NR02	Stop	Stop the video content	Native	Reproduction	10.8.2	2.11.6
NS01	Resize	Resize the viewports	Native	Size	10.5.8	1.8.3
NV01	Mute	Enable or disable the audio content	Native	Volume	10.6.2	1.5.1
NV02	Volume	Adjust the volume	Native	Volume	10.6.2	1.5.1
AR01	Rewind	Delay seconds within a reproduction	Additional	Reproduction	10.8.3	2.11.7
AR02	Forward	Forward seconds within a reproduction	Additional	Reproduction	10.8.3	2.11.7
AA01	Caption	Enable or disable captions	Additional	Alternatives	10.1.3,10.7.2	1.1.2
AA02	AudioDescription	Enable or disable the audio description	Additional	Alternatives	10.1.3	1.1.2
AA03	Size	Change the size of the captions	Additional	Alternatives	10.7.3	1.4.1
AA04	Font	Change the font of the captions	Additional	Alternatives	10.7.3	1.4.1
AA05	Color	Change the color of the captions	Additional	Alternatives	10.7.3	1.4.1
AA06	LanguageCaption	Change the language of the captions	Additional	Alternatives	8.2.1	2.7.1
AA07	LanguageAudio	Change the language of the audio description	Additional	Alternatives	8.2.1	2.7.1
AH01	Help	Documentation about accessibility features	Additional	Help	11.1.5	3.3.2
AF01	Find	Search within reproduction captions	Additional	Find		2.4.1

Among these recommendations, it is important to highlight that a user agent has to combine different alternatives for audiovisual information with video like captions, audio description, sign language, transcription and extended audio description, among others. Besides, it must provide complete access to all features using

mouse, keyboard or assistive technology (like screen readers). Moreover, it is necessary that this content provides help and documentation about its accessibility characteristics on the user interface, which reports the availability of those characteristics to the user, as well as the information about their purpose and use. Table 1 shows the accessibility requirements offered by the review grouped by categories. These categories are the result of abstracting the different requirements depending if the requirements are basics (native in Table 1), which group the typical requirements included in the user agent or media player or if the requirements add new functionalities (additional in Table 1), which are necessary to satisfy specific accessibility requirements.

As it is shown in Table 1, every requirement has a code in order to identify itself. This code is composed of the first letter of the group, the first letter of the subgroup and two numbers, for example, the requirement NR01 belongs to native group, reproduction subgroup and it has been assigned the number 01 within them.

Besides these requirements, there are other important requirements concerning accessibility close to usability. Among them, it is very important to keep accessibility features that are configured by users in following sessions and that those features only change when the user wants. It is essential to allow users to enable or disable and adjust accessibility features, as well as these features have to be easy to find and they have to be operable. On the other hand, keyboard shortcuts should allow navigating easily through the content, menus, submenus and lists and users must be informed about the available keyboard shortcuts. And last but not least, it is important to allow users to set their preferences to configure the keyboard shortcuts.

4. A conceptual approach of an accessible user agent

In order to design an accessible user agent that provides accessible video content with the requirements show in Section 3, the UsiXML technology has been selected. This UIDL is a declarative language capturing the essence of what a User Interface is or should be independently of physical characteristics. This language is structured according to the four levels of abstraction defined by the Cameleon reference framework [0]: Task and Concepts, Abstract User Interface (AUI), Concrete User Interface (CUI) and Final User Interface (FUI). UsiXML is deeply recognized and used to design and develop interactive systems, due to the richness of the offered models. In addition, it supports device, platform and modality independence. UsiXML describes the User Interface for multiple contexts of use such as Graphical User Interfaces (GUIs), Character User Interfaces (CUIs), Multimodal User Interfaces (MUI) and Auditory User Interfaces. Thus, interactive applications with different types of computing platforms, interaction techniques and modalities of use, can be described in a way that sustain the design independently from peculiar characteristics of physical computing platform.

4.1. Task Modeling

The accessibility requirements described in Section 3 have been modeled through the Task Model of UsiXML. This step is supported by a tool called IdealXML [0]. In Fig 1 is shown the interaction between a user and an accessible user agent taking into consideration how a user agent that provides accessible video content operates. There are also defined several tasks and some relationships between them.

In order to gather the semantics of interacting through playing, stopping, rewinding or forwarding the video reproduction, it has been defined an interaction task ‘Play’ (requirement NR01) and an abstract task that includes three interaction tasks: ‘Stop’, ‘Rewind’ and ‘Forward’ (requirements NR02, AR01 and AR02). These last tasks are related to each other or with themselves through temporal relationships. There are a temporal unary relationship, iteration, in task ‘Play’ and two temporal unary relationships, optional and iteration, in the others. This happens because task ‘Play’ has to be executed at least once, while the rest of the tasks can be executed or not (Guideline 2.11.2 of UAAG 2.0). On the other side, the temporal binary relationship between ‘Play’ and the abstract task is defined like enabling, it means, ’Play’ enables the others tasks when it finishes. Therefore, the user agent allows users to stop, rewind and forward the reproduction after finishing task ‘Play’.

The task ‘Resize’ (requirement NS01) is created to gather the semantics of enlarging the size of the user agent screen. Like other tasks, this also has two unary relationships (optional and iteration) with itself and an independent concurrency relationship with other tasks. In this case, the task related to enlarge the size of the viewport can be executed at any time if the user wishes to see the reproduction in a larger way and in any order.

The semantics of establishing the volume that means, increasing or decreasing the volume or getting the volume out is gathered defining two tasks: ‘Mute’ and ‘Volume’ (requirements NV01 and NV02). Both tasks have an optional and an iteration relationship with themselves and binary relationships of independent concurrency with other task. The execution of these tasks is similar to the previous tasks. In this case, the user agent has to allow users to modify the volume depending on their preferences at that moment.

The tasks ‘AudioDescription’ and ‘LanguageAudio’ (requirements AA02 and AA07) are contained within an abstract task. Both tasks have two temporal unary relationship, optional and iteration, with themselves and a temporal binary relationship between them, it means, the language of the audio description cannot be changed within the established values until the audio description is enabled and therefore, is considered finished.

Another abstract task ‘Caption’ is composed of a set of interactive and abstract tasks. This global task has the temporal binary relationship independent concurrency, so, it can be executed in any order. Besides that, this task presents three levels. The first level is composed of an interactive task ‘Caption’ (requirement AA01) that has two temporal unary relationships, optional and iterative, and an abstract task ‘ActionCaption’ that includes the second level of the global task. These tasks have a temporal binary relationship between them called enabling where the end of the task ‘Caption’ (that means, the enabling of this task) enables the other set of tasks of this level (the word search and the captions settings). The task ‘ActionCaption’ is composed of an interaction task ‘Find’ (requirement AF01, which allows user to seek words within the reproduction) that is optional and iterative and a set of tasks ‘Settings’. Both of them present an independent concurrency as a temporal binary relationship because they can be executed at any time. Within the group of settings, it can be found a set of optional and iterative tasks: ’Size’, ‘Font’, ‘Color’ and ‘LanguageCaption’ (requirements AA03, AA04, AA05 and AA06 respectively) and relationships of independent concurrency that allow users to change the size, color, font and language of the captions in any order.

The semantics of helping users is gathered through the interactive task ‘Help’ following the requirement AH01. Due to that, users can obtain some information about accessibility features that allows them to interact with the user agent in a properly and satisfactory way.

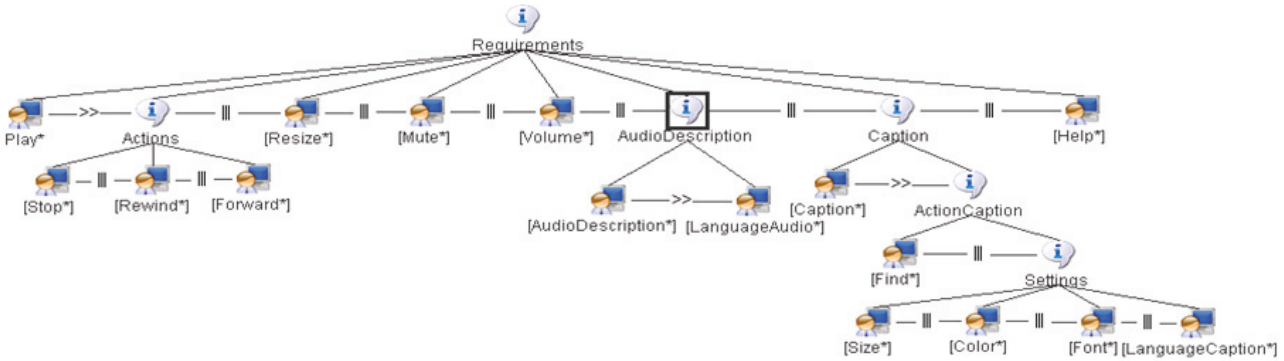

Fig 1. Task model of accessible user agent that provides accessible video content

4.2. Abstract User Interface Modeling

In another abstraction level, the semantics of the accessibility requirements is represented by the AUI of UsiXML. This step is also supported by IdealXML. The aim is to specify the abstract interaction objects (AIO)

and their relationship in order to design the user interface, therefore, this model shows the structure of the interaction elements. It is important to take into consideration that this model is accomplished starting from the task model presented in Fig 1 and described in Section 4.1.

An AUI is composed of Abstract Containers (AC), Abstract Individual Components (AIC) and abstract relationships between them. AIC represents basic system interactive functions: input, output, navigation and control. In this sense, AICs are an abstraction of widgets found in graphical toolkits (like windows, buttons) and in vocal toolkits (like vocal input and output widgets in the vocal interface). Therefore, our requirements are grouped in three ACs (that include AICs and some ACs) and five AICs.

As far as ACs is concerned, the first AC (Actions) is composed of three AICs (requirements NR02, AR01 and AR02). These requirements are composed of a control element, due to the functionality of these interaction objects that allows user to interact with the user agent through stopping, rewinding and forwarding without modifying any of its values. The second AC (AudioDescription) is composed of two AICs (requirements AA02 and AA07) because AA07 has to be executed after AA02 is enabled. Requirement AA02 is made up of a control element to allow users to listen to the audio description while requirement AA07 is made up of two elements, an input and an output. These elements allow users to select and establish a language within a list of languages from this time to the end of the reproduction or until the following change. The third AC (Caption) is composed of several ACs and AICs. Due to the execution order, it is defined an AIC (requirement AA01) that is made up of a control element to show captions without modifying any features of them. At the same level, it is defined an AC that is composed of an AIC (requirement AF01) and an AC where are grouped four AICs (requirements AA03, AA04, AA05 and AA06). Requirement AF01 presents an input, an output and a navigation element because the interaction is accomplished when the user establishes a word, the user agent seeks this word through captions and then it shows the corresponding caption within the reproduction. The rest of the caption characteristics (AA03, AA04, AA05 and AA06) are made up of two elements, an input and an output. It is due to the fact that the semantics of these characteristics can change the settings of the captions through selecting a value within a list of values as it has been mentioned before regarding requirement AA07.

Considering AICs, five AICs have been defined. Three requirements NR01, NS01and NV01 are composed of a control element. The functionality of these interaction objects allows users to interact with the user agent through playing, resizing or getting the volume out. On the other side, the functionality of requirement NV02 is different; in this case, users can modify the volume of the reproduction, so, this requirement needs an input element that allows user to interact with the user agent and to modify its current value. At least the requirement AH01 is composed of an input and a control element in order to allow users to select and show the help needed.

This composition is essential to follow the correct execution order where NR01 has to be executed first.

The semantics of the accessibility requirements indicated in Section 3 has been modeled. Following this, users can interact with an accessible user agent that provides accessible video content. The generated interactions are both only an action or an action and a modification (such as change the size of the captions).

The obtained abstract user interface design can be used to design different final user interfaces depending on several factors such as the context or the mode of use. Thus, a user interface can be described using UsiXML in a way that remains autonomous with respect to the devices used in the interactions (mouse, screen, etc.), to the various computing platforms (mobile phone, Pocket PC, etc.) and to any interaction modality (graphical interaction, vocal interaction, etc.). Besides including accessibility requirements described according to standards like WCAG 2.0, UAAG 2.0, the proposed design with the use of UsiXML technology provides the possibility to produce different interfaces depending on the end user profiles and context of use with the player. Therefore, this design approach provides a more universal and accessible solution.

4.3. Next steps in development process

UsiXML is used throughout the different steps of a Model Driven Engineering (MDE) compliant

development life cycle to store the models involved in the various processes. The following step in the presented abstract interface will be to pass to concrete levels like the CUI and finally, the FUI.

The CUI concretizes an AUI for a given context of use into Concrete Interaction Objects (CIOs) so as to define layout, the interface navigation of 2D graphical widgets and vocal widgets. Any CUI is composed of CIOs, which realize an abstraction of widgets sets found in popular graphical and vocal toolkits (e.g., Java AWT/Swing, HTML 5, Voice- XML, etc). For example, the AIC 'Volume' (requirement NV02) is transformed into a progressionBar element due to the fact that it presents an indeterminate attribute that depending on its value allows users to go from a minimum value to a maximum value or go from left to right and on the contrary. Other AC and AIC that present a control element ('Play', 'Stop', 'Resize' or 'Mute' among others) are transformed into button elements that trigger any kind of action available in the system.

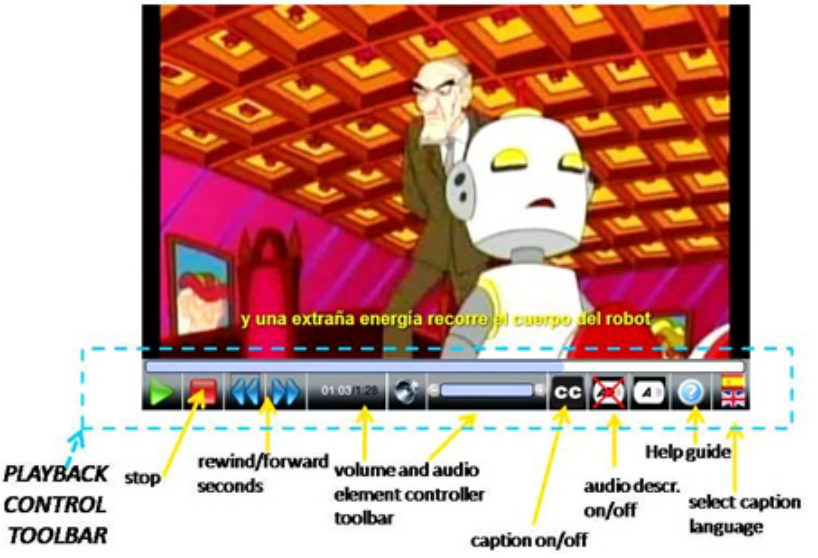

Fig 2. An accessible media player developed with the standard HTML5

The FUI is the operational user interface on a particular computing platform, e.g. through a Web browser and it is reached when the code of a user interface is produced from the previous levels seen before.

An example of the design proposal is shown in Fig 2. In this case, it is developed in Web platform using HTML5. The accessible user agent fulfills every accessibility requirement that has been defined in Section 3.

It is important to maintain the accessibility in the development process. Moreover, focusing on the accessibility requirements of the user agent and the video content; is essential that the accessibility requirements included in the first levels (Task and AUI) are maintained and transferred in the process. For this reason, the models and transformation rules of UsiXML technology must be revised. For example, it will be required the interaction elements of the abstract model like the control labels. It is necessary to transfer these labels through the development process until the FUI in order to satisfy the accessibility requirements in general, such as the assignation of a name for every element of the FUI or the possibility of visualizing the labels of the interaction elements in the same way that they are specified within the UAAG.

5. Conclusions

Because Web content is continuously growing, it is crucial that software could be accessible for anybody, especially software that provides video content and its access. For this reason, it is necessary to design an

accessible user agent that provides video content.

In order to get this aim, it is essential to consider different aspects such as the abstract elements that compose the user agent; the types of relationships that are going to establish among these elements or the concrete presentation of the user agent at the end of the design process. Therefore, this paper is accomplished to show how this type of user agent can be regarding accessibility features.

A set of accessibility requirements is proposed and an accessible user agent design to integrate into the development process is introduced. For this aim, UsiXML language has been used. Due to the fact that UsiXML supports platform, modality and device independence, it is a good choice for designing in a way that remains autonomous with respect to the devices used in the interactions (mouse, screen, etc.), to the various computing platforms (mobile phone, Pocket PC, etc.) and to any interaction modality (graphical interaction, vocal interaction, etc.). Every possibility helps to acquire more accessibility through WAI standards taking into consideration adaptive rapprochement according to user and device profiles.

Acknowledgements

This research work is supported by the Research Network MAVIR (S2009/TIC-1542 (www.mavir.net/)). We thank the European project ITEA UsiXML to provide the documentation of the UsiXML specification.

References

[1] W3C, WAI, Web Content Accessibility Guidelines (WCAG), 2010, http://www.w3.org/WAI/intro/wcag.php.

[2] W3C, Web Accessibility Initiative (WAI), 2010, http://www.w3.org/WAI/.

[3] W3C, Web Accessibility Initiative (WAI), http://www.w3.org/WAI/intro/components.php.

[4] W3C, WAI, User Agent Accessibility Guidelines (UAAG), 2010, http://www.w3.org/WAI/intro/uaag.php.

[5] User Interface Extensible Markup Language (UsiXML), http://www.usixml.org/.

[6] User Interface Description Languages (UIDL), http://www.uidl.net.

[7] W3C, Web Content Accessibility Guidelines 2.0, 2008, http://www.w3.org/TR/WCAG20/.

[8] Moreno L, Martínez P, Ruíz-Mezcua B. Disability Standards for Multimedia on the Web. *IEEE MultiMedia* 2008; 15(4): 52-54.

[9] IBM Web Accessibility Checklist, W3C Web Content Accessibility Guidelines (WCAG) and US Section 508, http://www-03.ibm.com/able/guidelines/web/ibm508wcag.html.

[10] WebAIM Section 508 Checklist, http://webaim.org/standards/508/checklist.

[11] W3C, Web Content Accessibility Guidelines 2.0 Checklist, 2005, http://www.w3.org/TR/2005/WD-WCAG20-20050630/checklist.

[12] Miyashita H, Takagi H, Sato D, Asakawa C, 2007. Making Multimedia Internet Content Accessible and Usable. Universal Access in Human-Computer Interaction. Applications and Services, 4th International Conference on Universal Access in Human-Computer Interaction, UAHCI 2007 Held as Part of HCI International 2007. Beijing, China, July 22-27, 2007 Proceedings, Part III.

[13] WebAIM, 2010, Media Player Accessibility, http://webaim.org/techniques/captions/mediaplayers/.

[14] Joe Clark, Accessibility, Design and Writing, 2011, http://joeclark.org/.

[15] Alfred P. Sloan Foundation, 2008, http://www.sloan.org/.

[16] NCAM, National Center for Accessible Media, 2009, http://ncam.wgbh.org/.

[17] Stanford Captioning, 2009, http://captioning.stanford.edu/.

[18] W3C, MediaAccessibility Úser Requirements, 2012, http://www.w3.org/TR/media-accessibility-reqs/.

[19] W3C, User Agent Accessibility Guidelines 1.0, 2002, http://www.w3.org/TR/WAI-USERAGENT/.

[20] W3C, User Agent Accessibility Guidelines 2.0, 2011, http://www.w3.org/TR/UAAG20/.

[21] ISO, International Organization for Standardization, 2008, http://www.iso.org.

[22] W3C, HTML5, A vocabulary and associated APIs for HTML and XHTML, 2011, http://www.w3.org/TR/html5/.

[23] Moreno L, González-García M, Martínez P, Iglesias A, 2011. A study of accessibility requirements for media players on the Web, 14th International Conference on Human-Computer Interaction (HCI 2011), 6th International Conference on Universal Access in Human-Computer Interaction. Orlando, Florida, USA, July, Springer Computer Science Editoria; Vol. LNCS 6765, pp 249-257.

[24] González-García M, Moreno L, Martínez P, Iglesias A, 2011. Requisitos de accesibilidad web en los reproductores multimedia, Interacción 2011, XII Congreso de Interacción Persona-Ordenador. Lisboa, Portugal, September; ISBM:978-84-9281-234-9, pp 43-53.

[25] González-García M, Moreno L, Martínez P, Iglesias A, 2011. Web accessibility requirements for media players, INTERACT 2011, 13th IFIP TC13 Conference on Human-Computer Interaction. Lisboa, Portugal, September; LNCS, Springer, Vol. 6949, pp 669-674.

[26] Calvary G, Coutaz J, Thevenin D, Limbourg Q, Bouillon L, Vanderdonckt J. A Unifying Reference Framework for Multi-Target User Interfaces. Interacting with Computers, June 2003; Vol. 15, No. 3, pp. 289-308.

[27] Montero F, Lopez VM. *Computer-Aided Design of User Interfaces V*, Springer, Holanda, 2007; pp 245-252.